1. A method comprising:

identifying a first data element to be removed from a data stream including other data elements;

writing the other elements into buffers and reading those elements from the buffers; and

preventing the first data element from being read from any of said buffers.

2. The method of claim 1 wherein identifying a first data element to be removed includes identifying the location of virtual local area network tags within the data stream.

3. The method of claim 1 wherein preventing the first data element from being read from any of said buffers includes preventing said first data element from being written to any of said buffers.

4. The method of claim 1 wherein preventing the first data element from being read from any of said buffers includes writing the first data element into a buffer and then overwriting said first data element in said buffer with one of said other data elements.

5. The method of claim 1 wherein writing the other elements into buffers includes writing the other elements

into buffers having a size comparable to the size of said first data element.

6. The method of claim 1 including producing a contiguous uninterrupted output data stream with said first data element removed.

7. The method of claim 1 including receiving a data stream including said first data element and other data elements and distributing said other data elements to a plurality of buffers.

8. The method of claim 7 including reading said data elements out of said buffers through a multiplexer to generate a contiguous data stream.

9. The method of claim 1 including receiving a data unit that includes two data elements, storing one of said two data elements in a first buffer and the other of said two data elements in a second buffer.

10. The method of claim 9 including outputting one of said two data elements through a first multiplexer and outputting the other of said data elements through a second multiplexer.

11. An article comprising a medium storing instructions that enable a processor-based system to:

identify a first data element to be removed from a data stream to include other data elements;

write the other elements into buffers and read those elements from the buffers; and

prevent the first data element from being read from any of said buffers.

12. The article claim 11 further comprising a medium storing instructions that enable a processor-based system to identify the location of virtual local area network tags within the data stream.

13. The article of claim 11 further comprising a medium storing instructions that enable a processor-based system to prevent said first data element from being written to any of said buffers.

14. The article of claim 11 further comprising a medium storing instructions that enable a processor-based system to write the first data element into a buffer and then overwrite said first data element in said buffer with one of said other data elements.

15. The article of claim 11 further comprising a medium storing instructions that enable a processor-based system to write the other elements into buffers having a size comparable to the size of said first data element.

16. The article of claim 11 further comprising a medium storing instructions that enable a processor-based system to produce a contiguous uninterrupted output data stream with said first data element removed.

17. The article of claim 11 further comprising a medium storing instructions that enable a processor-based system to receive a data stream to include said first data element and other data elements and distribute said other data elements to a plurality of buffers.

18. The article of claim 17 further comprising a medium storing instructions that enable a processor-based system to read said data elements out of said buffers through a multiplexer to generate a contiguous data stream.

19. The article of claim 11 further comprising a medium storing instructions that enable a processor-based system to receive a data unit that includes two data elements, store one of said two data elements in a first

buffer and the other of said two data elements in a second buffer.

20. The article of claim 19 further comprising a medium storing instructions that enable a processor-based system to output one of said two data elements through a first multiplexer and output the other of said data elements through a second multiplexer.

21. A system comprising:

a device to receive a plurality of data elements;

a plurality of buffers coupled to said device; and

a control to identify a first data element to be removed from a data stream to include other data elements, to write the other data elements into the buffers and read those elements from the buffers, and to prevent the first data element from being read from any of said buffers.

22. The system of claim 21 wherein said system is an Ethernet adapter.

23. The system of claim 21 wherein said system strips virtual local area network tags from said data stream.

24. The system of claim 21 wherein said control prevents the first data element from being read from any of said buffers.

25. The system of claim 21 wherein said control writes the first data element into a first buffer of said buffers and then overwrites the first data element in said first buffer with one of said other data elements.

26. The system of claim 21 wherein said buffers have a size comparable to the size of said first data element.

27. The system of claim 21 wherein said system produces a contiguous uninterrupted output data stream with said first data element removed.

28. The system of claim 21 including at least one multiplexer coupled to said buffers to store said other data elements.

29. The system of claim 28 including an output multiplexer coupled to said buffers to generate a contiguous data stream.

30. The system of claim 29 including a pair of output multiplexers, data units received by said device being

separated into a least two separated data units, said separated data units being output from different ones of said output multiplexers.

31. The system of claim 21 wherein the number of buffers equals the data clock size divided by the data size times the quantity of one plus the number of data elements to be removed.

32. The system of claim 21 wherein the number of buffers equals the data clock size divided by the data size times the quantity of two plus the number of data elements to be removed.